

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **62232036 A**

(43) Date of publication of application: **12 . 10 . 87**

(51) Int. Cl

G06F 9/38

G06F 15/16

(21) Application number: **61076981**

(71) Applicant: **NEC CORP**

(22) Date of filing: **02 . 04 . 86**

(72) Inventor: **IWASAKI TAMOTSU**

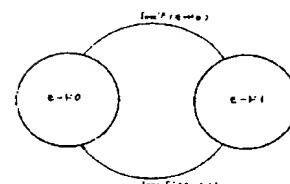
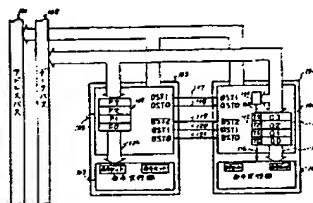
(54) INFORMATION PROCESSOR

COPYRIGHT: (C)1987,JPO&Japio

(57) Abstract:

PURPOSE: To eliminate necessity of correcting a large quantity of software so far used when adopting a new instruction set by making each instruction set include an alteration instruction that converts to execution of an instruction based on another instruction set.

CONSTITUTION: Two instruction sets are prepared respectively in instruction executing sections 107, 108 of a CPU103 and a coprocessor 104, and mode change instructions Inst.F, Inst.F' that instruct change of execution (mode 0) of an instruction based on the instruction set to execution (mode 1) of an instruction based on other instruction set are included in each instruction set of the CPU103. Instructions executed respectively by the CPU103 and coprocessor 104 constitute instruction queues 109, 110. The coprocessor 104 has a pre-decoder 111 that judges whether instructions fetched to constitute the instruction queue 10 are mode change instructions Inst.F, Inst.F' or not, and outputs an identification information [1] in the case of mode change instructions and outputs [0] in other cases.



⑯ 日本国特許庁 (JP)

⑮ 特許出願公開

⑯ 公開特許公報 (A)

昭62-232036

⑯ Int.Cl.¹

G 06 F 9/38
15/16

識別記号

370

府内整理番号

C-7361-5B
J-2116-5B

⑯ 公開 昭和62年(1987)10月12日

審査請求 未請求 発明の数 1 (全7頁)

⑯ 発明の名称 情報処理装置

⑯ 特願 昭61-76981

⑯ 出願 昭61(1986)4月2日

⑯ 発明者 岩崎保 東京都港区芝5丁目33番1号 日本電気株式会社内

⑯ 出願人 日本電気株式会社 東京都港区芝5丁目33番1号

⑯ 代理人 弁理士 内原晋

明細書

1. 発明の名称

情報処理装置

2. 特許請求の範囲

(1) 命令キューに格納された命令セットを順次実行可能な中央処理装置と、該中央処理装置と同期して同一命令のフェッチおよび実行を行なうコプロセッサとを含む情報処理装置において、上記中央処理装置が実行する複数の命令セットに対して各命令セットに該命令セットに基づく命令の実行から他の命令セットに基づく命令の実行に遷移させる変更命令を含ませたことを特徴とする情報処理装置。

(2) 上記コプロセッサは中央処理装置の有する命令セットに対応する複数の命令セットと、命令のフェッチ時に該命令が変更命令であるか否かを判別し上記フェッチされた命令が変更命令のときには識別情報を出力する命令識別情報を上

記フェッチされた命令に対応させて格納する識別情報格納手段とを有し、すでにフェッチされている命令の実行時に該命令に対応する識別情報を識別情報格納手段から読み出し命令セットの変更を行なうようにした特許請求の範囲第1項記載の情報処理装置。

3. 発明の詳細な説明

〔 産業上の利用分野 〕

本発明は情報処理装置。詳しくは、中央処理装置(以下、CPUといふ)に割当てられた命令キューと同じ大きさの命令キューを保持し、CPUから出力されるステータスによってCPUと同期してフェッチを行うと共に、CPUの有する命令セットに含まれる命令を実行するスレーブプロセッサ(以下、コプロセッサといふ)を含む情報処理装置に関する。

〔 従来の技術 〕

従来この種のコプロセッサはマスター・プロセッサと共に使用される。かかるCPUとコプロセッサ

特開昭62-232036 (2)

サとを使用したシステム例が第3図に示されており、この第3図のシステムで301と302とはそれぞれのアドレスバスとデータバスとを示している。CPU303のバスインターフェース部305とコプロセッサ304のバスインターフェース部306とはアドレスバス301とデータバス302を介して図示していないプログラムメモリおよびワーキングメモリと命令あるいはデータの授受を行なう。例えば、命令はCPU303の单一命令セットを有する命令実行部307とコプロセッサ304のCPUと対応する命令セットを有する命令実行部308とで実行されるのであるが、命令実行部307、308での実行に先立ち命令キュー309、310をそれぞれ形成し、内部データバス311、312をそれぞれ介して順次命令実行部307、308に供給される。313、314はCPU303の命令キュー309の状態（以下、キューステータスQST1、QST0という）をコプロセッサ304に送る信号線であり、315、316、317は起動しているバスサイクルの種類

を示す情報（以下、バスステータスBST2、BST1、BST0という）をCPU303からコプロセッサ304に送るための信号線である。

次に、CPU303がコプロセッサ304と共に動作して命令A～E（以下、Inst.A～Eという）を実行する手順を第4図に示された命令キューの状態図と別表1の解説を参照しつつ説明する。

第4図は、第3図に示したCPU303とコプロセッサ304の命令キューの内容と、CPU303、コプロセッサ304の状態及びCPU303が信号路305、306を通してコプロセッサ304に出力する情報と、同じく信号路307、308、309を通して出力する情報を時間を追って示している。簡単の為にキューは、4 Byteであり、各命令の実行に伴うメモリアクセスはないものとする。またInst.A～Eは1 Byte長の命令であり、Inst.C～EはCPU303の命令、Inst.Bはコプロセッサ304の命令である。

いまCPU303の命令キュー309とコプロセッサ304の命令キュー310には、Inst.A

～Dが入っている（第4図のステップ1）。

CPU303は、命令キュー309からInst.Aを取り出す。CPU303は、同時に信号路313、314を通してキュークリアをコプロセッサ304に送る。コプロセッサ304は、このキュークリアを受けて命令キュー310からInst.Aを取り出す。命令キューから取り出されたInst.Aは、CPU303では、内部データバス311を通して命令実行部307へ送られデコードされる。コプロセッサ304では、内部データバス312を通して命令実行部308へ送られデコードされる。Inst.Aは、CPU303の命令であり、各デコードの結果CPU303はInst.Aを実行し、コプロセッサ304はそのまま何も動作しない（第4図のステップ2）。

CPU303は、Inst.Aの実行が終了すると、命令キュー309からInst.Bを取り出す。同時にCPU303は、信号路313、314を通して命令コードの1 Byte目の取り出しを示すキュークリアによりInst.Bの実行と並行して命令キュー310からInst.Cを取り出す。Inst.Cは、CPU303の命令であるから、CPU303はInst.Cを実行する。コプロセッサ304は、Inst.Cのデコードによって何の影響も受けずInst.Bの実行をつづける（第4図のステップ4）。

ここでCPU303は、バスが使用されていた

命令キュー310からInst.Bを取り出す。Inst.Bは、Inst.Aと同様にCPU303の命令実行部307とコプロセッサ304の命令実行部308においてデコードされる。Inst.Bは、コプロセッサ304の命令であり、デコードの結果CPU303は何も動作せず、コプロセッサ304はInst.Bを実行する（第4図のステップ3）。

Inst.Bは、CPU303の命令ではないので、CPU303は、コプロセッサ304が命令を実行中であっても命令キュー309からInst.Cを取り出す。コプロセッサ304は、CPU303が出力する命令コードの1 byte目の取り出しを示すキュークリアによりInst.Bの実行と並行して命令キュー310からInst.Cを取り出す。Inst.Cは、CPU303の命令であるから、CPU303はInst.Cを実行する。コプロセッサ304は、Inst.Cのデコードによって何の影響も受けずInst.Bの実行をつづける（第4図のステップ4）。

ここでCPU303は、バスが使用されていた

特開昭62-232036 (3)

いことと、命令キュー-309内に命令コードが少なくなったことにより、次のようなフェッテ動作を行う。

CPU303は、命令キュー-309に入っている命令コードの次のアドレス(ここではInst.Bが格納されているアドレス)がアドレスバス301に出力される。次に出力されたアドレスに対応した命令コードがメモリからデータバス302上に出力される。CPU303は、このデータバス302に出力された命令コードを取り込み命令キュー-309に格納する。このフェッテ動作中、CPU303は、信号路315, 316, 317を通してフェッテ動作であることを示すバスステータスをコプロセッサ304に出力している。このステータスを受けてコプロセッサ304は、データバス302上の命令コードをCPU303が命令キュー-309に格納したのと同様にして命令キュー-310に格納する(第4図のステップ5)。以上の動作をステップ1~5ごとにまとめて示したのが別表1である。

る変更命令を含ませている。したがって、中央処理装置とコプロセッサとが同期して命令をフェッチし、中央処理装置がある命令セットに基づき命令キューを順次実行しているとき、上記変更命令が実行され、中央処理装置の命令セットが他の命令セットに切換えられると、以後中央処理装置は他の命令セットに基づき命令キュー中の命令を実行するようになる。したがって、中央処理装置の有する複数の命令セットに、新機能の実現を可能にする命令セットと従前の命令セットを含ませておけば、従前の命令セットに基づき開発されたソフトウェアを大幅に修正しなくともそのまま使用でき、しかも新機能の実現を可能にすることができる。

[実施例]

以下、本発明の一実施例を図面に基づき説明する。

第1図は一実施例の構成を示すブロック図であり、アドレスバス101、データバス102はCPU103とコプロセッサ104のバスインターフェース部105, 106とプログラムメモリ等の周辺装置とのアドレス情報、命令等の授受に使用されている。バスインターフェース部105, 106を介して供給される命令は命令実行部107, 108で命令セットに基づき実行される。CPU103とコプロセッサ104の命令実行部107, 108には2つの命令セットがそれぞれ用意されており、CPU103の各命令セットにはその命令セットに基づく命令の実行(モード0)から他の命令セットに基づく命令の実行(モード1)への変更を指示するモード変更命令Inst.F, Inst.F'が含まれている(第5図参照)。CPU103とコプロセッサ104とでそれぞれ実行される命令は命令キュー-109, 110を構成し、コプロセッサ104は命令キュー-110を構成するためにフェッチされる命令がモード変更命令Inst.F, Inst.F'であるか否かを判別しモード変更命令の場合は識別情報「1」を、それ以外の場合は「0」を出力するプリデコーダ111を有している。プリデコーダ111の出力は命令キュー-110に対

[発明が解決しようとする問題点]

一般に新規なCPUの開発に際しては、新機能の実現を図れる命令セットが必要であるが、従来開発された大量のソフトウェアをそのまま使用できるよう従来の命令セットを承継することも要請されている。ところが、上記従来のCPU303はコプロセッサ304と共通する命令を含む1つの命令セットしか有していないかったので、コプロセッサ304の支援を受けられるという利点に対して従前のCPUに関して開発されていた大量のソフトウェアの修正を必要とするという問題点があった。

従って、本発明は新作な命令セットの採用に際して従前のソフトウェアの修正を不要にできる情報処理装置に関する。

[問題点を解決するための手段、作用および効果]

本発明に係わる情報処理装置にあっては、中央処理装置が実行する複数の命令セットに対して、各命令セットに該命令セットに基づく命令の実行から他の命令セットに基づく命令の実行の遷移させ

ーフェース部105, 106とプログラムメモリ等の周辺装置とのアドレス情報、命令等の授受に使用されている。バスインターフェース部105, 106を介して供給される命令は命令実行部107, 108で命令セットに基づき実行される。CPU103とコプロセッサ104の命令実行部107, 108には2つの命令セットがそれぞれ用意されており、CPU103の各命令セットにはその命令セットに基づく命令の実行(モード0)から他の命令セットに基づく命令の実行(モード1)への変更を指示するモード変更命令Inst.F, Inst.F'が含まれている(第5図参照)。CPU103とコプロセッサ104とでそれぞれ実行される命令は命令キュー-109, 110を構成し、コプロセッサ104は命令キュー-110を構成するためにフェッチされる命令がモード変更命令Inst.F, Inst.F'であるか否かを判別しモード変更命令の場合は識別情報「1」を、それ以外の場合は「0」を出力するプリデコーダ111を有している。プリデコーダ111の出力は命令キュー-110に対

特開昭62-232036 (4)

応するビット群112に格納され、ビットT0～T3は命令Q0～Q3にそれぞれ対応している。114, 115は命令キューレコード109, 110内の命令を命令実行部107, 108にそれぞれ送出するための内部データバスであり、ビット群112に格納されていた情報「1」または「0」は、信号線116を介して命令と同期して命令実行部108に送出され、命令実行部108はビット群112から出力された情報に基づき命令セットの切換えを行なう。

次にCPU103がコプロセッサ104と共に動作して命令(Inst.B, Inst.F, Inst.B')をモードを切り換えつつ実行するときの作用を説明する。

第2図は、第1図に示したCPU103とコプロセッサ104の命令キューレコードの内容と、CPU103、コプロセッサ104の状態及びCPU103が信号路117, 118を通してコプロセッサ104に出力する情報と、同じく信号路119, 120, 121を通して出力する情報を時間を追って示している。簡単の為に各命令の実行に伴う

P2, P3に格納する。同様にコプロセッサ104も、Inst.F, Inst.B', Inst.C'を命令キューレコード110のQ1, Q2, Q3に格納する。ここでプリデコーダ111は、Inst.Fがフェッチされたことによって、命令実行モードを遷移させることを示す信号“1”を信号路113を通して、命令キューレコード110のInst.Fが格納されている場所に対応するビット群112のビット(すなわちT1)に格納する(第2図のステップ1)。

CPU103は、命令キューレコード109のP0からInst.Bを取り出す。コプロセッサ104では、CPU103が出力するキューステータスにより、命令キューレコード110のQ0からInst.Bが内部データバス115を通して、ビット群112のT0から“0”が信号路116を通して命令実行部108へ送られる。コプロセッサ104はモード0の状態であり、この命令を実行する。一方CPU103は、続いて命令キューレコード109のP1からInst.Fを取り出す(第2図のステップ2)。

Inst.Fは、CPU103の命令実行モードを

メモリアクセスはないものとする。また命令は1Byte長の命令であり、CPU103の命令のうち“”の付いた命令はモード1の命令であり、付いていない命令はモード0の命令であるものとする。なおInst.Bは、コプロセッサ104の命令である。

いま、CPU103及びコプロセッサ104は、モード0の状態であり、命令キューレコード109と、コプロセッサ104の命令キューレコードには空の状態であるとする。CPU103は、フェッチを行いInst.Bを命令キューレコード109のP0に格納する。コプロセッサ104は、CPU103が出力するバスステータスによってCPU103と同期してInst.Bをフェッチし、命令キューレコード110のQ0に格納する。このときプリデコーダ111は、“0”を信号路113を通して、命令キューレコード110のInst.Bが格納されている場所に対応するビット群112のT0に格納する。

引き続きCPU103は、Inst.F, Inst.B', Inst.C'をフェッチし、命令キューレコード109のP1、

遷移させる命令であり、CPU103では、この命令を実行することにより命令実行モードがモード0からモード1へと遷移する。同時にコプロセッサ104でも、命令キューレコード110のQ1からInst.Fが内部データバス115を通して、ビット群112のT1から“1”が信号路116を通して命令実行部108へ送られる。信号路116を通して命令実行部108へ送られた信号が“1”であることにより、コプロセッサ104でも、命令実行モードがモード0からモード1へと遷移する(第2図のステップ3)。

CPU103は、Inst.B'を命令キューレコード109のP2から取り出し実行する。コプロセッサ104でも、CPU103が出力するキューステータスに従って、Inst.B'を命令キューレコード110から取り出す。Inst.B'は、モード0の命令Inst.Bと同じコードであるが、コプロセッサ104の命令実行モードは、モード1となっているために実行は行わない(第2図のステップ4)。

ここでCPU103は、再びフェッチを行い、

Inst.F', Inst.C, Inst.D を命令キュー 109 の P0, P1, P2 に格納する。同様にコプロセッサ 104 も、Inst.F', Inst.C, Inst.D を命令キュー 110 の Q0, Q1, Q2 に格納する。プリデコーダ 111 は、フェッチされた Inst.F' を検出し、命令実行モードを遷移させる命令であることを示す信号 "1" を信号路 113 を通して、命令キュー 110 の Inst.F' が格納されている場所に対応するビット群 112 のビット（すなわち T0）に格納する（第2図のステップ 5）。

以下、コプロセッサ 104 は、CPU 103 と同期して命令を実行し、Inst.F' が命令キュー 110 から取り出される時に、同時に取り出されるビット群 112 の T0 の信号により再び命令実行モードをモード 1 からモード 0 へと遷移させる。以上の各ステップ 1～5 における CPU 103、コプロセッサ 104 等の機能をまとめたのが別表 2 であり、別表 3, 4 にはキューステータスとバスステータスとが説明されている。

〔効果の説明〕

以上説明したように、本発明によれば新たな命令セットの採用に際して、それまで用いられてきた大量のソフトウェアを修正する必要がないという非常に優れた効果を有している。

以下余白

別表 1

	ステップ 1	ステップ 2	ステップ 3	ステップ 4	ステップ 5
CPU 103 の状態	—	Inst.A の実行	—	Inst.C の実行	—
コプロセッサ 104 の状態	—	—	Inst.B の実行	Inst.B の実行	—
キューステータス	1 バイト目の取り出し	1 バイト目の取り出し	1 バイト目の取り出し	変化なし	1 バイト目の取り出し
バスステータス	受動	受動	受動	フェッチ	受動

別表 2

	ステップ 1	ステップ 2	ステップ 3	ステップ 4	ステップ 5
CPU 103 の状態	— (モード 0)	— (モード 0)	Inst.F' の実行 (モード 0)	Inst.B' の実行 (モード 1)	— (モード 1)
コプロセッサ 104 の状態	— (モード 0)	— (モード 0)	— (モード 0)	— (モード 1)	— (モード 1)
キューステータス	1 バイト目の取り出し	1 バイト目の取り出し	1 バイト目の取り出し	変化なし	1 バイト目の取り出し
バスステータス	受動	受動	受動	フェッチ	受動

別表 3

QST1	QST0	キューの状態
0	0	変化なし
0	1	命令コードの1 Byte目の取り出し
1	0	空
1	1	命令コードの2 Byte目以降の取り出し

別表 4

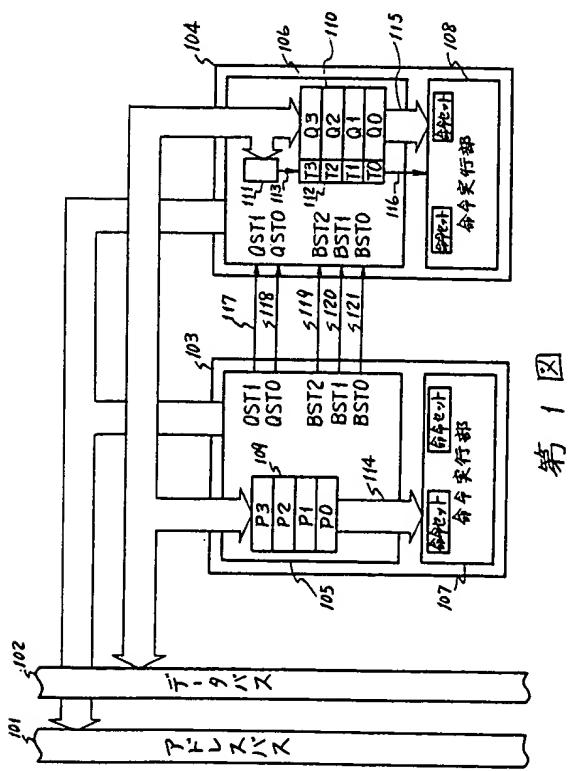
BST2	BST1	BST0	バスサイクロの意味
0	-	-	未使用
1	0	0	コードフェッチ
1	0	1	メモリリード
1	1	0	メモリライト
1	1	1	受動状態

4. 図面の簡単な説明

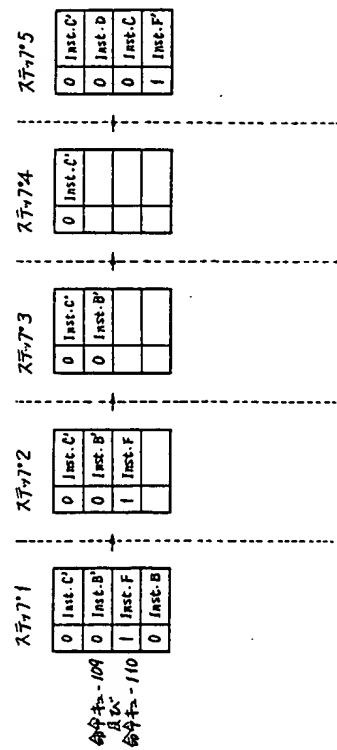
第1図は一実施例の構成を示すブロック図、第2図は一実施例の命令キューの状態図、第3図は従来例のブロック図、第4図は従来例の命令キューの状態図、第5図は命令実行モードの変更を示す概念図である。

1 0 3 ……中央処理装置、1 0 4 ……コプロセッサ、Inst.F, Inst.F' ……変更命令、1 1 1 ……命令識別手段(ブリデコーダ)、1 1 2 ……識別情報格納手段(ビット群)。

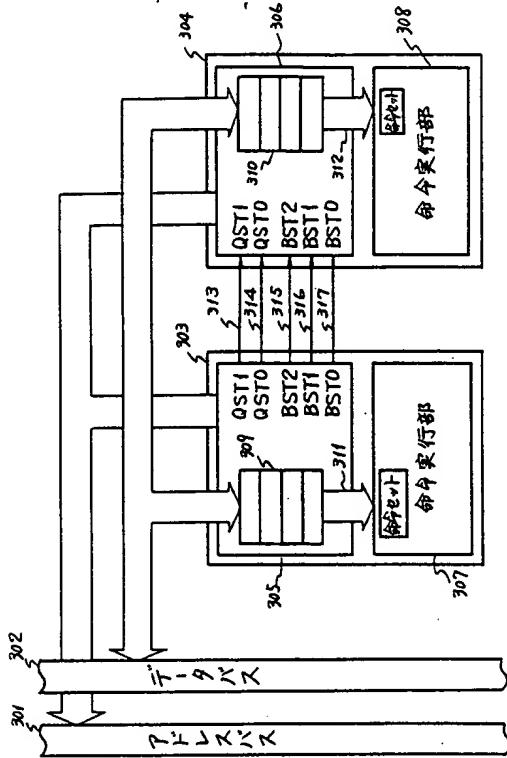
代理人 弁理士 内原 錠



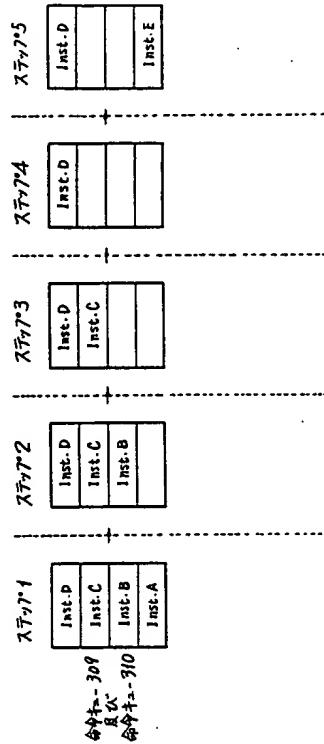
第1図



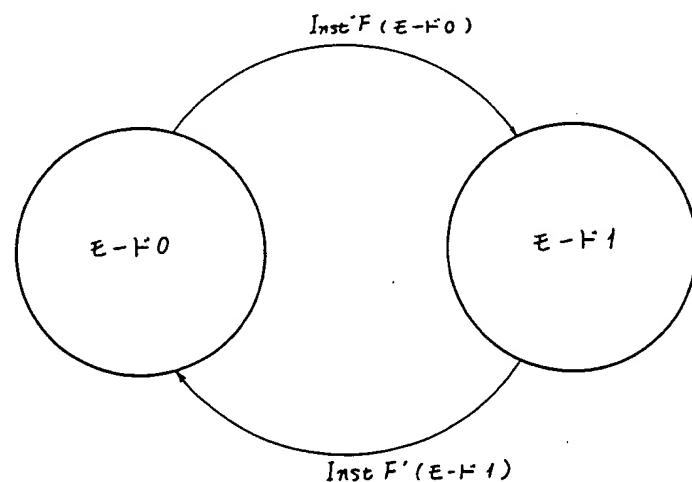
第2図



第3図(従来例)



第4図(従来例)



第5図